IN THE CLAIMS

Please amend the claims as follows:

Claim 1 (Currently Amended): A processor power-saving control method which employs a plurality of Operating Systems (OSs) whose execution is controlled by a processor, wherein said plurality of OSs include

a primary OS for receiving a timer interrupt issued from a hardware timer after a predetermined time lapse, and

a secondary OS treated as a task to be executed by said primary OS, said processor power-saving control method comprising the steps of:

~~upon~~ receiving said timer interrupt at said primary OS[[,]];

determining with said primary OS whether there exists any task scheduled to be executed on said secondary OS; ~~and~~

~~when the primary OS determines there exists any task to be executed on said secondary OS,~~ interrupting said secondary OS by issuing a secondary-OS interrupt from the primary OS to the secondary OS when the primary OS determines there exists any task scheduled to be executed on said secondary OS; and

removing the secondary OS from a power-saving mode in response to the secondary OS receiving the secondary OS-interrupt issued from the primary OS.

Claim 2 (Currently Amended): A processor power-saving control method which employs a plurality of Operating Systems (OSs) whose execution is controlled by a processor,

operation of said processor being ~~stopped~~ placed in a power-saving mode when there exists no task scheduled to be executed on said plurality of OSs,

said processor power-saving control method controlling timer interrupt processing performed by a hardware timer which ~~activates~~ causes said processor to exit the power-saving mode after an arbitrary time lapse,

said processor power-saving control method keeping a power-saving state of said processor,

said plurality of OSs including a primary OS for receiving a timer interrupt issued from said hardware timer and a secondary OS treated as a task to be executed by said primary OS,

said processor power-saving control method comprising:

a primary-OS process step performed by said primary OS; a secondary-OS process step performed by said secondary OS; and a secondary-OS interrupt step,

said primary-OS process step including

detecting said timer interrupt,

a first determination step of, upon receiving said timer interrupt, determining whether there exists any task scheduled to be executed, and

a processor stopping step of~~, when there is no task to be executed, stopping~~ placing said processor in the power-saving mode when there is no task scheduled to be executed;

said secondary-OS process step including

a second determination step of determining whether there exists any task scheduled to be executed, and

~~when there is no task to be executed,~~ handing over processing to said first determination step when there is no task scheduled to be executed;

said secondary-OS interrupt step including

receiving a secondary-OS interrupt from said primary OS,

~~when said first determination step has determined that there exists any task to be executed on said secondary OS,~~ performing interrupt processing on said secondary OS when said first determination step has determined that there exists any task scheduled to be executed on said secondary OS,, and

executing said second determination step at a predetermined time measured from said interrupt.

Claim 3 (Original): The processor power-saving control method as claimed in claim 2, wherein said secondary-OS interrupt step is executed by a periodically-activating handler which interrupts said secondary OS at regular time intervals.

Claim 4 (Original): The processor power-saving control method as claimed in claim 2, wherein said secondary-OS interrupt step is executed by an alarm handler which interrupts said secondary OS after a specified time period.

Claim 5 (Currently Amended): The processor power-saving control method as claimed in claim 2, wherein said secondary-OS interrupt step is executed by a high-priority task which is a task for interrupting said secondary OS and has a highest priority order among tasks scheduled to be executed by said primary OS.

Claim 6 (Original): The processor power-saving control method as claimed in claim 2, wherein said processor stopping step includes a step of determining whether time taken until said hardware timer issues a next timer interrupt is longer than a predetermined time, and if a measured time is longer than said predetermined time, said processor stopping step stops operation of said processor.

Claim 7 (Currently Amended): The processor power-saving control method as claimed in claim 2, wherein said primary-OS process step further comprises steps of:

when said hardware timer periodically performs timer interrupt processing at regular time intervals, determining whether timer interrupt processing is required again by a time at which a task is scheduled to be executed; and

if timer interrupt processing is not required again, stopping said hardware timer.

Claim 8 (Original): The processor power-saving control method as claimed in claim 7, wherein said primary-OS process step further comprises a step of:

detecting a timer interrupt issued by a long-periodic hardware timer which issues a timer interrupt at a time interval longer than that of said hardware timer.

Claim 9 (Original): The processor power-saving control method as claimed in claim 2, wherein said primary-OS process step further comprises a step of:

detecting a timer interrupt issued by a time-of-day timer which measures a time of day as well as issuing a timer interrupt at a predetermined time of day.

Claim 10 (Currently Amended): A computer readable storage medium storing a plurality of Operating Systems (OSs) whose execution is controlled by a processor which is stopped when there exists no task scheduled to be executed, said plurality of OSs including a primary OS for receiving a timer interrupt issued from a hardware timer which activates said processor after an arbitrary time lapse, and a secondary OS treated as a task to be executed by said primary OS,

wherein said computer readable storage medium stores a program which causes a computer to perform: a primary-OS process step on said primary OS[[,]]; a secondary-OS process step on said secondary OS[[,]]; and a secondary-OS interrupt step,

said primary-OS process step including

detecting [[a]] said timer interrupt issued by said hardware timer,

a first determination step of, upon detecting said timer interrupt, determining whether there exists any task scheduled to be executed, and

a processor stopping step of~~, when there is no task to be executed, stopping~~ placing said processor in a power-saving mode where there is no task scheduled to be executed;

said secondary-OS process step including

a second determination step of determining whether there exists any task scheduled to be executed, and

~~when there is no task to be executed,~~ handing over processing to said first determination step when there is no task scheduled to be executed;

said secondary-OS interrupt step including

receiving a secondary-OS interrupt from said primary OS,

~~when said first determination step has determined that there exists any task to be executed on said secondary OS,~~ performing interrupt processing on said secondary OS when said first determination step has determined that there exists any task scheduled to be executed on said secondary OS, and

executing said second determination step at a predetermined time measured from said interrupt.

Claim 11 (Currently Amended): A processor power-saving control device comprising:

a timer ~~means~~ device including a hardware timer ~~for issuing~~ configured to issue a timer interrupt after an arbitrary time lapse, and ~~activating~~ activate a processor, operation of said processor being ~~stopped~~ placed in a power-saving mode when there exists no task scheduled to be executed; and

a storage ~~means for storing~~ device configured to store a primary Operating System (OS) and a secondary OS;

wherein said primary OS, upon receiving said timer interrupt at said primary OS, is configured to determine whether there exists any task scheduled to be executed, and to ~~stop~~ cause said processor to enter the power-saving mode if there is no task scheduled to be executed and to issue a secondary-OS interrupt if there is any task scheduled to be executed; and

said primary OS, upon receiving the secondary-OS interrupt from said primary OS, is configured to determine whether there exists any task scheduled to be executed, and if there is any task scheduled to be executed, to execute the task, said secondary OS being treated as a task scheduled to be executed by said primary OS.

Claim 12 (Currently Amended): The processor power-saving control device as claimed in claim 11, wherein said timer ~~means~~ device has a long-periodic timer which issues a timer interrupt at a time interval longer than that of said hardware timer.

Claim 13 (Currently Amended): The processor power-saving control device as claimed in claim 11, wherein said timer ~~means~~ device has a time-of-day timer.

Claim 14 (Canceled).

Claim 15 (Currently Amended): The process power-saving control method of claim 2, wherein the secondary-OS interrupt step further includes the step of:

activating the secondary OS from ~~a sleep~~ the power-saving mode in response to the secondary OS receiving the secondary-OS interrupt from the primary OS.

Claim 16 (Currently Amended): The computer readable storage medium of claim 10, wherein the secondary-OS interrupt step includes the step of:

activating the secondary OS from ~~a sleep mode~~ the power-saving mode in response to the secondary OS receiving the secondary-OS interrupt from the primary OS.

Claim 17 (Currently Amended): The processor power-saving control device as claimed in claim 11, wherein said secondary OS is further configured to activate from ~~a sleep mode~~ the power-saving mode in response to the receipt of the secondary-OS interrupt.

Claim 18 (New): The method of claim 1, wherein removing the secondary OS from the power-saving mode further comprises increasing a frequency of a processor clock signal.

Claim 19 (New): The method of claim 1, wherein removing the secondary OS from the power-saving mode further comprises exiting a sleep mode using a sleep Application Program Interface (API).

Claim 20 (New): The method of claim 2, wherein placing said processor in the power-saving mode further comprises reducing a frequency of a processor clock signal.

Claim 21 (New): The method of claim 2, wherein placing said processor in the power-saving mode further comprises stopping a processor clock signal.

Claim 22 (New): The method of claim 2, wherein placing said processor in the power-saving mode further comprises entering a sleep mode using a sleep Application Program Interface (API).

Claim 23 (New): The computer readable storage medium of claim 10, wherein placing said processor in the power-saving mode further comprises reducing a frequency of a processor clock signal.

Claim 24 (New): The computer readable storage medium of claim 10, wherein placing said processor in the power-saving mode further comprises stopping a processor clock signal.

Claim 25 (New): The computer readable storage medium of claim 10, wherein placing said processor in the power-saving mode further comprises entering a sleep mode using a sleep Application Program Interface (API).

Claim 26 (New): The device of claim 11, wherein the primary OS is further configured to cause the processor to enter the power-saving mode by reducing a frequency of a processor clock signal.

Claim 27 (New): The device of claim 11, wherein the primary OS is further configured to cause the processor to enter the power-saving mode by stopping a processor clock signal.

Claim 28 (New): The device of claim 11, wherein the primary OS is further configured to cause the processor to enter the power-saving mode by entering a sleep mode using a sleep Application Program Interface (API).

Claim 29 (New): The method of claim 15, wherein activating the secondary OS from the power saving mode further comprises increasing a frequency of a processor clock signal.

Claim 30 (New): The method of claim 15, wherein activating the secondary OS from the power saving mode further exiting a sleep mode using a sleep Application Program Interface (API).

Claim 31 (New): The method of claim 16, wherein activating the secondary OS from the power saving mode further comprises increasing a frequency of a processor clock signal.

Claim 32 (New): The method of claim 16, wherein activating the secondary OS from the power saving mode further exiting a sleep mode using a sleep Application Program Interface (API).

Claim 33 (New): The device of claim 17, wherein the secondary OS is further configured to activate from the power-saving mode by increasing a frequency of a processor clock signal.

Claim 34 (New): The device of claim 17, wherein the secondary OS is further configured to activate from the power-saving mode by exiting a sleep mode using a sleep Application Program Interface (API).